CS 383

Lecture 01 – Introduction

Stephen Checkoway

Fall 2023

What is CS 383 all about?

This is a very mathematical course with a lot of practical applications

The overarching theme is computation

Three parts to the course

1. Automata (singular is automaton) (8 weeks)
2. Computability (4 weeks)
3. Complexity (3 weeks)

Computation I

One main theme of this course is what can be computed and what can't

Which problems can be solved by computers and which can't

Here are some problems we can solve with computers

- Sort a finite list
- Check if an integer is prime
- Draw some triangles on a screen
- Determine the shortest path between two vertices (nodes) in a graph
- Factor a polynomial
- Never lose at tic-tac-toe
- Never lose at checkers (solved in 2007 but took 18 years!)
- ...

Computation II

Everything you've learned in CS so far has been about solving problems

When you see a new problem, you might think,
"I know how to solve this problem"; or
"I don't know how to solve it right now, but I'm sure I can figure it out"; or maybe
"Eventually, someone will figure out how to solve it"

Computation II

Everything you've learned in CS so far has been about solving problems

When you see a new problem, you might think,
"I know how to solve this problem"; or
"I don't know how to solve it right now, but I'm sure I can figure it out"; or maybe
"Eventually, someone will figure out how to solve it"

Here are some problems we know we can't solve with computers

- Given a computer program, will the program crash when run on some input?
- Given two computer programs, do they compute the same answer when given the same input for all inputs?
- Given a multivariable, polynomial equation, determine if it has a solution in integers
- Find the cheapest airfare between two airports (this is surprisingly complicated)
- Lots of problems in mathematics
- ...

Decision problems

In this course, we're going to focus on problems whose answers are Yes/No (or True/False)

These are decision problems

Examples

- Is an integer n even?
- Does a directed graph G have a path of length n between vertices u and v?
- Does a program P crash when run on input x?
- Is x an element of a set S?
- Does a string s end with a repeated letter?

Models of computation

Real computers are frighteningly complex

They are much too complicated to reason about

Instead, we're going to focus on simpler models of computation

Finite automaton used in text-processing and compilers
Context-free grammar used in programming languages and compilers
Turing machines equivalent in power to general purpose computers

The models are progressively more powerful; they let us solve more problems

Administrivia

Course web page: `https://checkoway.net/teaching/cs383/2023-fall/`
(also linked from Blackboard)

Textbook: Michael Sipser's *Introduction to the Theory of Computation* 3rd. edition
(the 2nd. edition is fine too!)

Ed discussion forum: All communication with course staff must be done via Ed unless you have been explicitly instructed otherwise

Grades

- Homework (60%)
- Two midterm exams (20%)
- Final exam (20%)

You are free to work with other students on assignments but you must write them up individually

Exams are open book/notes

Late policy

Homework is due at 23:59

You have 3 late days to use throughout the semester

No need to ask me; you're responsible for keeping track of how many you've used

Homework policy

You may collaborate with other students in the class on the homework

You must write up your solutions entirely on your own!

Your write up must be typeset, not hand written

I strongly encourage you to use LaTeX, but you're free to use other tools

See the course web page for LaTeX resources and examples

Academic misconduct policy

Academic misconduct is taken very seriously

All incidents will be reported as honor code violations

Penalties range from a 0 on an assignment to failing the course

Academic misconduct includes (but is not limited to)

- Using generative AI/large language models (e.g., ChatGPT and Bard)
- Using someone else's solutions as your own
- Searching the Internet for solutions to homework
- Copying answers on tests
- Showing someone else your solutions on tests
- Deceiving course staff (e.g., giving a false excuse for missing a deadline)
- Using material for exams that hasn't been explicitly authorized (e.g., the book, notes, slide print outs)

Large language models don't know anything

I asked ChatGPT,
Prove that the square root of 8 is irrational.

It responded:
The square root of 8 is not irrational, it is actually a rational number. The square root of 8 can be simplified as the square root of 4 times 2, which is equal to 2 times the square root of 2. And since the square root of 2 is irrational, 2 times the square root of 2 is equal to 2 times an irrational number, which is still an irrational number. So, the square root of 8 is equal to 2 times an irrational number and is therefore a rational number.

Just don't use it

Sets I

A set is a collection of objects (numbers, symbols, other sets, anything)

The objects in the set are the elements or members

E.g., $S = \{2, 3, 5, 7, 11\}$ is a 5-element set

We use $\in$ and $\notin$ to denote set membership and nonmembership

$5 \in S$ and $9 \notin S$

Common sets:

The empty set is written $\varnothing$

The set of natural numbers is written $\mathbb{N} = \{1, 2, \ldots\}$

The set of integers is written $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

The set of rational numbers is written $\mathbb{Q}$

The set of real numbers is written $\mathbb{R}$

Sets II

We can define sets by giving rules for sets

$$\textsc{Primes} = \{x \mid x \text{ is a prime number}\}$$
$$\textsc{Odds} = \{2x + 1 \mid x \in \mathbb{Z}\}$$
$$T = \{n \mid n = m^2 \text{ for some } m \in \mathbb{N}\}$$

Set A is a subset of B (written $S \subseteq B$) if every element of A is an element of B
Set A is a proper subset of B (written $S \subsetneq B$) if $A \subseteq B$ and $A \neq B$

Set operations

Union $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$; elements of either A or B

Set operations

Union $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$; elements of either A or B

Intersection $A \cap B = \{x \mid x \in A \text{ and } x \in B\}$; elements of both A and B

Set operations

Union $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$; elements of either A or B

Intersection $A \cap B = \{x \mid x \in A \text{ and } x \in B\}$; elements of both A and B

Complement $\overline{A} = A^{\mathsf{C}} = \{x \mid x \notin A\}$; elements not in A

Set operations

Union	$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$; elements of either A or B
Intersection	$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$; elements of both A and B
Complement	$\overline{A} = A^{\complement} = \{x \mid x \notin A\}$; elements not in A
Difference	$A \setminus B = \{x \mid x \in A \text{ and } x \notin B\} = A \cap \overline{B}$; elements of A but not B

Set operations

Union	$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$; elements of either A or B
Intersection	$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$; elements of both A and B
Complement	$\overline{A} = A^{\complement} = \{x \mid x \notin A\}$; elements not in A
Difference	$A \setminus B = \{x \mid x \in A \text{ and } x \notin B\} = A \cap \overline{B}$; elements of A but not B
Power set	$P(A) = \{S \mid S \subseteq A\}$; set of subsets of A

Set examples

$$A = \{0, 1, 2, 3, 4\}$$
$$B = \{2, 4, 5\}$$
$$C = \{3x \mid x \in \mathbb{Z}\}$$

Set examples

$$A = \{0, 1, 2, 3, 4\}$$
$$B = \{2, 4, 5\}$$
$$C = \{3x \mid x \in \mathbb{Z}\}$$

$$A \cup B =$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$A \cup B = \{0, 1, 2, 3, 4, 5\}$$

Set examples

$$A = \{0, 1, 2, 3, 4\}$$
$$B = \{2, 4, 5\}$$
$$C = \{3x \mid x \in \mathbb{Z}\}$$

$$A \cup B = \{0, 1, 2, 3, 4, 5\}$$
$$A \cap B =$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\}
\end{aligned}
$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &=
\end{aligned}
$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\}
\end{aligned}
$$

Set examples

$$A = \{0, 1, 2, 3, 4\}$$
$$B = \{2, 4, 5\}$$
$$C = \{3x \mid x \in \mathbb{Z}\}$$

$$A \cup B = \{0, 1, 2, 3, 4, 5\}$$
$$A \cap B = \{2, 4\}$$
$$A \cap C = \{0, 3\}$$
$$B \cap C =$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing
\end{aligned}
$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &=
\end{aligned}
$$

Set examples

$$\begin{aligned} A &= \{0, 1, 2, 3, 4\} \\ B &= \{2, 4, 5\} \\ C &= \{3x \mid x \in \mathbb{Z}\} \end{aligned}$$

$$\begin{aligned} A \cup B &= \{0, 1, 2, 3, 4, 5\} \\ A \cap B &= \{2, 4\} \\ A \cap C &= \{0, 3\} \\ B \cap C &= \varnothing \\ A \setminus B &= \{0, 1, 3\} \end{aligned}$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &= \{0, 1, 3\} \\
P(A \setminus B) &=
\end{aligned}
$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &= \{0, 1, 3\} \\
P(A \setminus B) &= \big\{\varnothing, \{0\}, \{1\}, \{3\}, \{0, 1\}, \{0, 3\}, \{1, 3\}, \{0, 1, 3\}\big\}
\end{aligned}
$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &= \{0, 1, 3\} \\
P(A \setminus B) &= \big\{\varnothing, \{0\}, \{1\}, \{3\}, \{0, 1\}, \{0, 3\}, \{1, 3\}, \{0, 1, 3\}\big\} \\
P(\varnothing) &=
\end{aligned}
$$

Set examples

$$\begin{aligned} A &= \{0,1,2,3,4\} \\ B &= \{2,4,5\} \\ C &= \{3x \mid x \in \mathbb{Z}\} \end{aligned}$$

$$\begin{aligned} A \cup B &= \{0,1,2,3,4,5\} \\ A \cap B &= \{2,4\} \\ A \cap C &= \{0,3\} \\ B \cap C &= \varnothing \\ A \setminus B &= \{0,1,3\} \\ P(A \setminus B) &= \big\{\varnothing, \{0\}, \{1\}, \{3\}, \{0,1\}, \{0,3\}, \{1,3\}, \{0,1,3\}\big\} \\ P(\varnothing) &= \{\varnothing\} \end{aligned}$$

Set examples

$$\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}$$

$$\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &= \{0, 1, 3\} \\
P(A \setminus B) &= \big\{\varnothing, \{0\}, \{1\}, \{3\}, \{0, 1\}, \{0, 3\}, \{1, 3\}, \{0, 1, 3\}\big\} \\
P(\varnothing) &= \{\varnothing\} \\
P\big(P(\varnothing)\big) &=
\end{aligned}$$

Set examples

$$
\begin{aligned}
A &= \{0, 1, 2, 3, 4\} \\
B &= \{2, 4, 5\} \\
C &= \{3x \mid x \in \mathbb{Z}\}
\end{aligned}
$$

$$
\begin{aligned}
A \cup B &= \{0, 1, 2, 3, 4, 5\} \\
A \cap B &= \{2, 4\} \\
A \cap C &= \{0, 3\} \\
B \cap C &= \varnothing \\
A \setminus B &= \{0, 1, 3\} \\
P(A \setminus B) &= \big\{\varnothing, \{0\}, \{1\}, \{3\}, \{0, 1\}, \{0, 3\}, \{1, 3\}, \{0, 1, 3\}\big\} \\
P(\varnothing) &= \{\varnothing\} \\
P\big(P(\varnothing)\big) &= \big\{\varnothing, \{\varnothing\}\big\}
\end{aligned}
$$

Tuples

Tuples are finite sequences of objects in some order

$(2, 7, 8)$

$(\mathtt{a}, \mathtt{a}, \mathtt{b}, \mathtt{a}, \mathtt{b})$

$\big(\varnothing, \{0, 1\}, \{0\}\big)$

Order of elements in a tuple matters, unlike in a set

Repeated elements in a tuple matter, unlike in a set

A tuple with k elements is called a k-tuple

A pair is a 2-tuple

Cartesian product

A Cartesian product of two sets is defined by

$$A \times B = \{(x, y) \mid x \in A \text{ and } y \in B\}$$

A Cartesian product of k sets, A_1 through A_k, is defined by

$$A_1 \times A_2 \times \cdots \times A_k = \{(x_1, x_2, \ldots, x_k) \mid x_i \in A_i\}$$

If $A = \{\texttt{a}, \texttt{b}\}$ and $B = \{1, 2, 3\}$, then

$$A \times B = \{(\texttt{a}, 1), (\texttt{a}, 2), (\texttt{a}, 3), (\texttt{b}, 1), (\texttt{b}, 2), (\texttt{b}, 3)\}$$

We can take a repeated Cartesian product of a set with itself k times

$$A^k = \underbrace{A \times A \times \cdots \times A}_{k}$$

Functions

Functions are mappings of elements from one set, the domain, to another set, the range

- The range is also called the codomain
- Some texts use range for a related, but distinct concept

We write $f : X \to Y$ where

- f – name of the function
- X – domain
- Y – range

To each element $x \in X$, f assigns exactly one element $y \in Y$, written $y = f(x)$

When the function f is clear (or unnamed), we can express that mapping as $x \mapsto y$ (read "x maps to y")

Function examples

- $inc : \mathbb{N} \to \mathbb{N}$ given by $n \mapsto n + 1$ (equiv. $inc(n) = n + 1$)

Function examples

- $inc : \mathbb{N} \to \mathbb{N}$ given by $n \mapsto n + 1$ (equiv. $inc(n) = n + 1$)
- $f : \mathbb{Z} \times \mathbb{R} \to \{\mathrm{true}, \mathrm{false}\}$ given by

$$f(n, x) = \begin{cases} \mathrm{true} & \text{if } |n - x| < 3 \\ \mathrm{false} & \text{otherwise} \end{cases}$$

Function examples

- $inc : \mathbb{N} \to \mathbb{N}$ given by $n \mapsto n + 1$ (equiv. $inc(n) = n + 1$)
- $f : \mathbb{Z} \times \mathbb{R} \to \{\text{true}, \text{false}\}$ given by

$$f(n, x) = \begin{cases} \text{true} & \text{if } |n - x| < 3 \\ \text{false} & \text{otherwise} \end{cases}$$

- $g : \{q_0, q_1, q_2\} \times \{\texttt{a}, \texttt{b}\} \to \{q_0, q_1, q_2\}$ given by the table

q	x	$g(q, x)$
q_0	a	q_1
q_0	b	q_0
q_1	a	q_2
q_1	b	q_1
q_2	a	q_0
q_2	b	q_2

Equivalently,

$$g(q_i, \texttt{a}) = q_{i+1 \bmod 3}$$
$$g(q_i, \texttt{b}) = q_i$$

Function examples

- $inc : \mathbb{N} \to \mathbb{N}$ given by $n \mapsto n + 1$ (equiv. $inc(n) = n + 1$)
- $f : \mathbb{Z} \times \mathbb{R} \to \{\mathrm{true}, \mathrm{false}\}$ given by

$$f(n, x) = \begin{cases} \mathrm{true} & \text{if } |n - x| < 3 \\ \mathrm{false} & \text{otherwise} \end{cases}$$

- $g : \{q_0, q_1, q_2\} \times \{\mathtt{a}, \mathtt{b}\} \to \{q_0, q_1, q_2\}$ given by the table

q	x	$g(q, x)$
q_0	a	q_1
q_0	b	q_0
q_1	a	q_2
q_1	b	q_1
q_2	a	q_0
q_2	b	q_2

Equivalently,

$$g(q_i, \mathtt{a}) = q_{i+1 \bmod 3}$$
$$g(q_i, \mathtt{b}) = q_i$$

Function examples

- $inc : \mathbb{N} \to \mathbb{N}$ given by $n \mapsto n + 1$ (equiv. $inc(n) = n + 1$)
- $f : \mathbb{Z} \times \mathbb{R} \to \{\text{true}, \text{false}\}$ given by

$$f(n, x) = \begin{cases} \text{true} & \text{if } |n - x| < 3 \\ \text{false} & \text{otherwise} \end{cases}$$

- $g : \{q_0, q_1, q_2\} \times \{\mathtt{a}, \mathtt{b}\} \to \{q_0, q_1, q_2\}$ given by the table

q	x	$g(q, x)$
q_0	a	q_1
q_0	b	q_0
q_1	a	q_2
q_1	b	q_1
q_2	a	q_0
q_2	b	q_2

Equivalently,

$$g(q_i, \mathtt{a}) = q_{i+1 \bmod 3}$$
$$g(q_i, \mathtt{b}) = q_i$$

Note that this completely specifies all 6 cases

Function examples II

- Function of 3-tuples: $h : \mathbb{N}^3 \to \mathbb{N}^2$ given by $h(x, y, z) = (3x + z, yz)$

Function examples II

- Function of 3-tuples: $h : \mathbb{N}^3 \to \mathbb{N}^2$ given by $h(x, y, z) = (3x + z, yz)$
- Bad example removed from slides, sorry! The key point was that a function $f_1 : X \to Y$ is different from a function $f_2 : P(X) \to Y$. The first takes a single element of X as an argument whereas the second takes a *set* of elements from X (a subset of X) as an argument.

Functions in CS 383

We're going to see a lot of functions that look like

$$\delta_1 : Q \times \Sigma \to Q$$
$$\delta_2 : Q \times \Sigma \to P(Q)$$

where Q and Σ are (finite) sets

δ_1 maps a pair (q, a) to an element of Q

δ_2 maps a pair (q, a) to a set of elements of Q

Functions later in CS 383

Later, we'll see a lot of functions that look like

$$\delta_3 : Q \times \Sigma \times \Gamma \to P(Q \times \Gamma)$$
$$\delta_4 : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

where Γ is some other set

δ_3 maps a triple (q, a, b) to a set of pairs, e.g., $\delta_3(q, a, b) = \{(r, c), (s, d)\}$

δ_4 maps a pair (q, a) to a triple, e.g., $\delta_4(q, a) = (r, b, L)$

Alphabets, strings, and languages

Alphabets, strings, and languages are the key building blocks for this course

Almost everything in the course boils down to asking the question: Is the string s an element of the language L?

Alphabets and symbols

An alphabet is a nonempty, finite set

The members of an alphabet are the symbols of the alphabet

We (usually) denote alphabets with the capital Greek letters Σ and Γ (and various subscripts)

$$\begin{aligned} \Sigma_1 &= \{\mathtt{0}, \mathtt{1}\} \\ \Sigma_2 &= \{\mathtt{a}, \mathtt{b}, \mathtt{c}\} \\ \Gamma &= \{\#, \mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{x}, \mathtt{y}\} \end{aligned}$$

Alphabets and symbols

An alphabet is a nonempty, finite set

The members of an alphabet are the symbols of the alphabet

We (usually) denote alphabets with the capital Greek letters Σ and Γ (and various subscripts)

$$\begin{aligned}\Sigma_1 &= \{\mathtt{0}, \mathtt{1}\}\\ \Sigma_2 &= \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}\\ \Gamma &= \{\#, \mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{x}, \mathtt{y}\}\end{aligned}$$

I will try my best to follow Sipser and write symbols in `typewriter font`

Strings

A string (also called a word) is a finite, possibly empty sequence of symbols from a given alphabet

- 0110110 is a string *over* the alphabet $\Sigma_1 = \{0, 1\}$
- aababacab is a string over the alphabet $\Sigma_2 = \{a, b, c\}$
- $2100\#xxy$ is a string over the alphabet $\Gamma = \{\#, 0, 1, 2, x, y\}$

The empty string is a sequence of zero symbols and is denoted ε

The length of a string w, written $|w|$ is the number of symbols it contains

- $|0110110| = 7$
- $|aababacab| = 9$
- $|2100\#xxy| = 8$
- $|\varepsilon| = 0$

String concatenation

We can concatenate two strings to produce a new string

- If x = aab and y = ba, then xy = aabba
- Concatenating ε does not change the string: $x\varepsilon = \varepsilon x = x$
- If x and y are strings, then $|xy| = |x| + |y|$
- If x is a string and k is a nonnegative integer, then $x^k = \underbrace{xx\cdots x}_{k}$ and $|x^k| = k \cdot |x|$

Substrings

A string s is a substring of w if all of the symbols in s appear consecutively in w

- 000 is a substring of 001000
- 0100 is a substring of 001000
- 0000 is not a substring of 001000

String reversal

If $w = w_1 w_2 \cdots w_n$ is a string of length n where $w_i \in \Sigma$, then $w^{\mathcal{R}} = w_n w_{n-1} \cdots w_1$ is the reversal of w

- $\texttt{abb}^{\mathcal{R}} = \texttt{bba}$
- $\texttt{a}^{\mathcal{R}} = \texttt{a}$
- $\varepsilon^{\mathcal{R}} = \varepsilon$

String prefix

String x is a prefix of string y if there exists string z such that $xz = y$

A string of length n has $n + 1$ prefixes

Prefixes of `aaba`

1. ε
2. `a`
3. `aa`
4. `aab`
5. `aaba`

ε has exactly one prefix: ε itself

String x is a proper prefix of string y if x is a prefix of y and $x \neq y$

Languages

A language is a (possibly infinite) set of strings over an alphabet Σ

- $L_1 = \varnothing$. The empty language
- $L_2 = \{\varepsilon\}$. The language containing only the empty string
- $L_3 = \{\mathtt{a}, \mathtt{aa}, \mathtt{aba}\}$
- $L_4 = \Sigma^*$. The language of all strings (remember, strings have finite length)
- $L_5 = \Sigma^+$. The language of all nonempty string ($L_5 = L_4 \setminus L_2$)
- $L_6 = \{\mathtt{a}^n\mathtt{b}^n \mid n \geq 0\} = \{\varepsilon, \mathtt{ab}, \mathtt{aabb}, \mathtt{aaabbb}, \dots\}$

Languages L_1, L_2 and L_3 are finite (meaning they have finitely many elements)

Languages L_4, L_5, and L_6 are infinite

Operations on languages

Languages are sets, so the usual set operations like union and intersection have the normal meanings

The complement of a language L is the set of all strings over the alphabet that are not in L. In symbols $\overline{L} = \Sigma^* \setminus L$

We'll see lots of operations defined on languages throughout the course

- Reversal. $L^{\mathcal{R}} = \{w^{\mathcal{R}} \mid w \in L\}$
- Composition (or concatenation). $L_1 \circ L_2 = \{xy \mid x \in L_1 \text{ and } y \in L_2\}$
- Kleene star. $L^* = \{x_1 x_2 \cdots x_k \mid k \geq 0 \text{ and each } x_i \in L\}$
- ...

Recap

Alphabets are finite, nonempty sets of symbols
E.g., $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, $\Gamma = \{\mathtt{0}, \mathtt{1}, \ldots, \mathtt{9}\}$

Strings are finite sequences of symbols from an alphabet
E.g., ε (the empty string), aab, $\mathtt{b}^5\mathtt{aab} = \mathtt{bbbbbaab}$

Languages are (possibly infinite) sets of strings
E.g., $\varnothing$, $\{\mathtt{a}\}$, Σ^*, $\{w \mid |w| \geq 3\}$

Question 1

Are the sets $\varnothing$ and $\{\varnothing\}$ the same?

Question 1

Are the sets $\varnothing$ and $\{\varnothing\}$ the same?

No. $\varnothing$ is the set containing no elements. $\{\varnothing\}$ is the set containing a single element, namely the empty set.

Question 2

If S is a set, then $\varnothing \subseteq S$. True or false?

Question 2

If S is a set, then $\varnothing \subseteq S$. True or false?

True. $\varnothing$ is a subset of every set

Question 3

Let $S = \{1, 2, 3\}$. Is $1 \subseteq S$?

Question 3

Let $S = \{1, 2, 3\}$. Is $1 \subseteq S$?

No. 1 is not a set and so it certainly cannot be a subset of S

Question 4

Let $S = \{1, 2, 3\}$. Is $\{2\} \in S$?

Question 4

Let $S = \{1, 2, 3\}$. Is $\{2\} \in S$?

No. $\{2\}$ is a set but S doesn't contain any sets

However, $\{2\} \subseteq S$

Question 5

Let $S = \{1, 2, 3\}$. Is $S \subseteq S$

Question 5

Let $S = \{1, 2, 3\}$. Is $S \subseteq S$

Yes. Every set is a subset of itself

Question 6

Is $\{0\}$ a valid alphabet?

Question 6

Is $\{0\}$ a valid alphabet?

Yes. It's called a unary alphabet because it has one symbol

Question 7

Is $\{0, 1\}$ a valid alphabet?

Question 7

Is $\{0, 1\}$ a valid alphabet?

Yes. It's called a binary alphabet because it has two symbols

Question 8

Is $\varnothing$ a valid alphabet?

Question 8

Is $\varnothing$ a valid alphabet?

No. Alphabets must be nonempty

Question 9

Is $\{0, 1, 2, \dots\}$ a valid alphabet?

Question 9

Is $\{0, 1, 2, \dots\}$ a valid alphabet?

No. Alphabets must have finitely many elements

Question 10

Is the sequence of symbols $\mathtt{a}\#\mathtt{b}$ a valid string over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$?

Question 10

Is the sequence of symbols $\texttt{a}\#\texttt{b}$ a valid string over the alphabet $\Sigma = \{\texttt{a}, \texttt{b}\}$?

No. All symbols in the string must come from the alphabet

It is a valid string over the alphabet $\Sigma' = \{\texttt{a}, \texttt{b}, \#\}$.

Question 11

Is the empty-length sequence of symbols ε a valid string over the alphabet $\Sigma = \{\mathtt{a}\}$?

Question 11

Is the empty-length sequence of symbols ε a valid string over the alphabet $\Sigma = \{\mathtt{a}\}$?

Yes. The empty string is a string over any alphabet

Question 12

Is the infinite-length sequence of a symbols, $\mathtt{aa}\ldots$, a valid string over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$?

Question 12

Is the infinite-length sequence of a symbols, $\mathtt{aa}\ldots$, a valid string over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$?

No. Strings must be finite length

Question 13

Can a language have zero elements?

Question 13

Can a language have zero elements?

Yes. $\varnothing$ is a perfectly reasonable language

Question 14

Can a language have infinitely many elements?

Question 14

Can a language have infinitely many elements?

Yes. Σ^*, the set of all strings over Σ is finite for any alphabet Σ

Consider $\Sigma = \{a\}$, then the language $\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$

Most of the languages in this course will be infinite

Question 15

Every language L of strings over the alphabet Σ is a subset of Σ^*. True or false?

Question 15

Every language L of strings over the alphabet Σ is a subset of Σ^*. True or false?

True. Σ^* is the set of all strings over Σ and L is some set of strings over Σ, so every element of L is an element of Σ^* and thus $L \subseteq \Sigma^*$

Next time

Read chapter 0 and section 1.1 of Sipser

We're going to talk about our first model of computation: deterministic finite automaton (DFA)

Components of a DFA:

- A finite set of states Q
- An input alphabet Σ
- A transition function $\delta : Q \times \Sigma \to Q$ that controls how the DFA moves from state to state on a given input
- A start state $q_0 \in Q$
- A set of accept states $F \subseteq Q$ that control whether or not the DFA *accepts* an input string